Orange

Automated AI model deployment within ONAP

Alassane Samba
Nicolas Edel
Philippe Dooze
Eric Debeau
AI for network automation

- ONAP will enable network automation
- Automated decisions and actions could be taken using Closed Loop
- AI will bring more intelligent decisions thanks to data
Forecasting based resource allocation

- Exploit seasonality and trend from data history
- Take into account exogenous variables (e.g. holidays) into account
- Get “one step ahead” thanks to forecasting
Challenges

Be able, without too much effort, to

- Build a new AI model compatible with ONAP
- Deploy an AI model in ONAP whatever its type
- Manage model lifecycle
  - Take into account new features
  - Enhance the algorithm, etc.
Experience

Design time

Steps:

1. Data collection
   a. preceded by the development and the deployment of a Data collection microservice

2. Model coding, training & evaluation

3. Model Transformation into Execution microservice using Acumos followed by
   a. Development of an I/O microservice dedicated to
      - reading model arguments on DMaaP
      - querying analysis/prediction from model
      - writing results on DMaaP
   b. Deployment of “Execution” & “I/O” microservices
Experience

Runtime

AI Model execution microservice

AI Model I/O microservice

Training platform

Datalake

ONAP

DMaaS

VES Collector

DCAE

Policy

APP-C

Deployed current ONAP component

Deployed from specific development

Model microservice

Other deployed component

Online training results

Training results

Training data

Model parameters

Store VES data and results

results

results

results

results

Event data

Event data

Event data

actions

decisions

Deployed current ONAP component

Deployed from specific development

Model microservice

Other deployed component
**Takeaways**

**Requirements:**
- Be careful about AI model characteristics: possible need of Datalake and/or Training platform
- Need of the combo “Execution” + “I/O” microservices at runtime
- Need to separate model parameters and training algorithm for online models

**New challenges:**
- Model I/O microservice generation tool
- Automated microservices generation directly from code repository: CI/CD for models
- Use of DCAE blueprint for automatic deployment of any new AI model

**Model characteristics:**

<table>
<thead>
<tr>
<th>Training mode</th>
<th>Training computation</th>
<th>Parameters</th>
<th>Argument passing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch</td>
<td>Massive</td>
<td>Embedded</td>
<td>Explicit</td>
</tr>
<tr>
<td>Online</td>
<td>Moderate</td>
<td>Remote</td>
<td>Semi-implicit</td>
</tr>
</tbody>
</table>

- Online training mode: need to store model parameters in a remote place from the model microservice
- Online-Massive training: need of a Datalake and a Training platform
- Implicit or Semi-implicit argument passing or Remote model parameters: need of a Datalake
Thank you

Welcome to the demo on the Orange booth!